

MERN Stack Lectureflow

The below given flow should be followed by each faculty while taking lectures. If the faculty decides to change the flow - he/she will need to first take permission from the Training coordinator at the HO (Ahmedabad office)

Module-2) SE - Overview of IT Industry	5
<ul style="list-style-type: none"> • Introduction of students • Career in IT • Understanding Student Login of TOPS ERP • Using Lab • What is Program • What is programming? • Types of Programming Language • World Wide Web • How Internet Works • Network Layers on Client and Server • Client And Servers • Types of Internet Connections • Protocols • Application Security • Software Applications and its types • Software Architecture • Layers in Software Architecture • Software Environments • Types of Programming Languages • Source Code • Github and introductions • Student Account in Github • Types of Software • Introduction of Software • Application software • Software development process • Software Requirement • Software Analysis • System Design • Software Testing • Maintenance • Development • Web Application • Designing • mobile application • DFD • Desktop Application • Flow Chart 	
Module 6) Mernstack - HTML	10

<ul style="list-style-type: none"> • 1) HTML Introduction 2) HTML Getting Started 3) HTML Elements 4) HTML Attributes 5) HTML Basic Tags • 1) HTML Doctypes 2) HTML Layout 3) HTML Head 4) HTML Meta 5) HTML Scripts • Practical Examples: 1) Create any simple web page to display your name. 2) Importance of meta tag and Doctypes • Tags and self Closing Tags, Basic Tag , Attribute and Events, Marquee Tag • HTML - Meta Tags, HTML - Comments, HTML - Images, HTML - Tables, HTML - Lists, HTML - Text Links, HTML - Image Links • HTML Headings HTML Paragraphs HTML Links HTML Text Formatting HTML Styles HTML Images • HTML - Frames, HTML - Iframes, HTML - Blocks, HTML - Backgrounds, HTML - Colors, HTML - Fonts • Anchor Tag, Img Tag, Image Mapping • HTML - Fonts, HTML - Forms, HTML - Embed Multimedia ,HTML - Marquees, HTML - Header, HTML - Style Sheet, HTML - Javascript ,HTML - Layouts • List Tag, Tables, Forms • PRactical Examples: 1) Create simple Doc and display your name using different heading tag 2) Create link for open google. 3) Create document using all text formatting tags • HTML Tables HTML Lists HTML Forms HTML Iframes • Practical Examples: 1) Create simple table 2) Create time table for your school 3) Create table with colspanrowspan example 4) Create invoice using table 5) Create hotel menu. 6) Create index page for your book. 7) Create list with different categories. • PRactical Examples: Create registration form with all fields and validation 	
Module 7) Mernstack - CSS and CSS 3	10
<ul style="list-style-type: none"> • 1) CSS 2) In-line CSS Internal Style External Style Sheet @import Style Sheet 3) CSS Class CSS ID • What is CSS How to Implement CSS Class and ID Width and Height Css Unit Box Model (Margin,padding,Border) and create basic template design • Practical example : Create page with difference color text • CSS Selectors , Pseudo Classes and Elements , Float and Clear and Alignment , Font Styling , Opacity and Visibility , Line Height • 1) CSS Text 2)CSS Font 3) CSS Background 4) CSS Links 5) CSS Lists 6) CSS Display 7) CSS Visibility • PRactical Example : Create layout for your project • Media Query (For Responsive Website) 	
Module 8) Mernstack - HTML5	2
<ul style="list-style-type: none"> • HTML5 Tags, HTML5 Input and Attribute • Audio and Video, Semantic Element in HTML5 • Canvas, Svg • Display Grid • Introduction of jQuery, jQuery Effects, jQuery Traversing 	
Module 12) Mernstack- Bootstrap Basic & Advanced	9

- Bootstrap Basic 1) Bootstrap Introduction 2) Bootstrap Getting Started 3) Bootstrap Grid System 4) Bootstrap Fixed Layout 5) Bootstrap Fluid Layout 6) Bootstrap Responsive Layout
- Practical Example: Create Navigation Menu
- 1) Bootstrap Typography 2) Bootstrap Tables 3) Bootstrap Lists 4) Bootstrap List Groups 5) Bootstrap Forms 6) Bootstrap Custom Forms 7) Bootstrap Input Groups 8) Bootstrap Buttons 9) Bootstrap Button Groups
- Practical Example: Create login registration form
- 1) Bootstrap Images 2) Bootstrap Cards 3) Bootstrap Media Objects 4) Bootstrap Icons 5) Bootstrap Navs 6) Bootstrap Navbar 7) Bootstrap Breadcrumbs 8) Bootstrap Pagination 9) Bootstrap Badges 10) Bootstrap Progress Bars 11) Bootstrap Spinners 12) Bootstrap Jumbotron 13) Bootstrap Helper Classes
- Practical Example: 1) Create image gallery 2) Create model for login Product list page with pagination
- Bootstrap Advanced 1) Bootstrap Modals 2) Bootstrap Dropdowns 3) Bootstrap Tabs 4) Bootstrap Tooltips 5) Bootstrap Popovers 6) Bootstrap Alerts 7) Bootstrap Stateful Buttons 8) Bootstrap Accordion 9) Bootstrap Carousel 10) Bootstrap Typeahead 11) Bootstrap ScrollSpy 12) Bootstrap Toasts
- Practical Example: Create your project website using bootstrap

Module 13) JavaScript Essentials And Advanced

25

- Basic JavaScript, Js comment, Js variables , Understanding var, let and Const, JS switch, if, else, JS loop , Js global variables, Js data types, Js operators, Js Functions
- Functions - Function Declaration in JS - Arrow Functions - Higher Order Functions - Map, Reduce and Filter
- Javascript Objects, Js object , Js Array , Js string, Js Date, Js Math, Js number, Js Boolean
- Javascript BOM ,Browser Objects , Window object, History object, navigator object, Screen object
- Javascript DOM, Document object, getElementById, getElementByName, getElementByTagName, JS innerHTML property, JS innerTEXT property
- Javascript OOPS, JS class, JS object, JS prototype, JS constructor method, JS static method, JS encapsulation, JS inheritance, JS polymorphism, JS abstractions
- Javascript Exception Handling, JS exception handling , Javascript try-catch
- Javascript MISC, JS this keyword , JS Debugging , JS Hoisting , JS Strict Mode, JS promises, JS typeof , JS ternary operator, JS reload() method, JS setAttributes () method, JS setInterval() method, JS setTimeout() method.
- Javascript Events, Javascript Events, Javascript AddEventListener(), jsOnClick event, jsdbclick event, JS onload event, JS onresize event.
- Array in JS, Creating Array, Array methods, The Spread & Rest operators, Destructuring
- JS Async, Callbacks, Promises, Async/Await
- ES6 Basics and Babel, New features in ES 6, Arrow functions, The . Operator, For/of , Map Objects, Set Objects, Promises, Functions Rest parameter, String.includes(),String.starts.With(), String.endsWith(), Array.form(), Array.keys(), Array find(), Array findIndex(), javascript Modules
- Small Project using ES6

Module-14) React - Components, State, Props

8

- Installation - Add React to a HTML Website - Create New React App - Hello World
- •Different ways to installation - Create first project in react - Add React to a HTML Website - Create New React App - Hello World
- Getting started in React
- •React Js Introduction •Project Structure Overview
- JSX
- •React Dom - Render HTML •SPA vs Traditional Web app
- Components
- Component Composition
- •Convert HTML to JSX
- JSX - Why JSX? - Embedding Expressions in JSX - Attributes with JSX - Children with JSX
- Props & Prop Types
- Types of Components, Difference between Function Components and Class components
- Event Handlers
- Class Lifecycle
- State
- Import - Export Components
- React Web App
- Working with Multiple components in same file, components nesting and composition
- Components, State, Props - Function Component - Class Component - Props - State - Class Component Lifecycle
- Conditional Rendering
- •Ternary , if, if..else , switch case
- State and its importance
- •Commonly used events (onclick, onchange..)
- •Event Handlers
- •useState - Hook introduction
- •Setstate in class components
- •Practical : accept value from user and display in h1 , create calculator app using state , hide - unhide div on button click
- •Prop and prop types
- •Updating Array in state
- •Updating object in state
- Basic understanding of Css - style

Module 15) Lists , Hooks , Localstorage , Api Project

6

- Conditional Rendering - Lists and Keys - Forms - Handling Events - Lifting State up
- •Rendering List in React
- Hooks - Introduction - Using the State hook - Using the Effect hook - Rules of Hook - Custom Hook
- •Use of Map() function
- Rendering Lists inside components
- •Keys and importance in lists
- React Keys
- •useRef
- Using keys wit component
- •React Refs
- Uniqueness of keys among siblings
- •Uses of React Refs
- React refs
- •useEffect practical
- Uses of react Refs
- •useContext
- How to access of Refs
- •useReducer
- Refs current properties
- •useCallback
- Add Refs to DOM elements
- •useMemo
- Add refs to class components
- •Custom hook
- Callback refs
- •Forms and Form handling
- Forwarding Ref from one component to another component
- •Form submission and validation
- React with useRef
- •React Hook form
- React conditional rendering
- •Project : Perform CRUD operations using array or map , render all records from array or map and display in table apply edit and delete operations – don't use any databases
- React if, logical & operator, Ternary operator, switch case operator, Conditional Rendering with Enum, Preventing components from rendering
- •JSON.parse and JSON.
- •Project : Perform CRUD operations using local storage database
- Project : Task manager application , pending task , completed task , select all , deselect all functionality

Module-16) React -Advance React- Styling , Routing

5

- Creating the first App
- React bootstrap
- Understanding the App
- Tailwind in React
- Styling the App
- Material UI in react js
- Inspecting & Debugging styles
- Advantages of material ui in react js
- Built-in components
- material UI components implementation
- Working with Images
- Buttons, Grid system , stack , typography , icons
- ListViews
- Customize style using material ui theme
- TextInput
- Styling React Components - CSS stylesheet - Inline Styling - CSS Modules - CSS in JS Libraries (styled components)
- AppBar
- Layout components
- Creating Views (Scenes)
- Drawer , grid layout
- Conditional Rendering - Lists and Keys - Forms - Handling Events - Lifting State up
- Form Components
- Hooks - Introduction - Using the State hook - Using the Effect hook - Rules of Hook - Custom Hook
- Advance Concepts - Context, useContext() - Working with Refs and useRefs() - Fragments - Performance optimization with useMemo() - Styling React Components - CSS stylesheet - Inline Styling - CSS Modules - CSS in JS Libraries (styled components)
- Textfields , checkboxes , Form validation
- Bootstrap with React
- Dialogs and modals
- React Router - Browser - Router - Link - Route - Template integration - Http Request in React - Get and Post data
- React Router - Browser - Router - Link , navlink
- Route Parameters - useParams
- Nested Routes
- Programmatic Navigation - history
- Lazy loading (performance optimization)
- Authentication - authorization
- Redirecting unauthorized used to login page
- Using css transitions or animation library like React transition group.
- Error Handling - 404 Page not found
- Project : Project using API Or FakeJsonAPI
- Api testing in Postman

Module 17) React - JSON-server and Firebase Real Time Database

<ul style="list-style-type: none"> • React Router • Installing Firebase SDK in react project • Browser - Router - Link - Route • Need of react router • NoSql Database in firebase • Creating collection and documents • Template integration - Http Request in React - Get and Post data • React router installation • Authentication with firebase • React router, react-router-native, react-router-Dom • Use of firebase storage • Component in react router , Browser Router , HashRouter • Project : Create app which contain registration , login and profile updation using firebase , user can post image and other user's can like dis-like post • What is Route • Project2: Web app using JSON-server • What is Link component , Adding navigation using Link component • Link vs NavLink • React Router Switch , React Router redirect • Nested Routing in React • Template integrations Using Browser Router , Routes , Route , Link and Hash Router • Advantages of react Router 	
Module-18) React - Applying Redux	8
<ul style="list-style-type: none"> • State • State storage problem • Redux Basics • Redux Principles • Implementing Redux • React-Redux • Redux Core concepts - Actions , Reducers , Store React-Redux • Middleware • Counter App Demo • Redux - Complexity of Managing state - Understand the Redux Flow - Setting up Reducer and store - Dispatching Actions - Passing and Retrieving Data with Action - Combining Multiple Reducers - Adding Middleware - Redux Dev tools • Projects - To Do App, News App, Tic Tac Toe, MEdium Clone, Redux Implementation, Writing Custom Redux Middleware, Instagram clone • Redux-thunk middleware • Project : Perform CRUD operations using React-Redux 	
Module 21) Node - NodeJS - Introduction	5
<ul style="list-style-type: none"> • Introduction to Node.js and its features, Understanding the basics of web servers and HTTP requests • Node Modules - HTTP, File Systems, etc., Understanding the Node.js event loop and asynchronous programming • Using callbacks, promises, and async/await in Node.js, Handling errors and debugging Node.js applications, • Implementing security best practices in Node.js 	
Module 19) Database - No SQL- Mongodb	6

<ul style="list-style-type: none"> • Introduction of MongoDB, No SQL Database, Advantage over RDBMS, MongoDB Data Types, Install MongoDB, MongoDB Data Modeling, • MongoDB Operators - Query & Projection Operator, MongoDB Update Operator, Aggregation Pipeline Stages, MongoDB limit() MongoDB sort() Query Modifiers Database, Commands, Aggregation Commands, Geospatial Command, Query and Write Operation Commands Query Plan Cache Commands Authentication Commands, User Management Commands, Role Management Commands, Replication Command, Sharding Commands, Session Commands, • Database - Create Database, Drop Database • Collection - Create Collection, Drop Collection CRUD : Documents Inset Documents, Update Documents, Delete Documents, Query Documents, SQL to MongoDB Mapping, MongoDB text search, Partial Updates & Document, Limits, Removing Documents, Multi Update, Upsert, Wire Protocol, Bulk() Operations and Methods, Cursor Method, Query Plan Cache Methods, User Management Method, Role Management Method, MongoDB Replication Methods, • MongoDB Cloud - MongoDB Atlas, MongoDB Tools, MongoDB Compass, 	
Module 20) Introduction to Express JS	5
<ul style="list-style-type: none"> • Introduction to Express Framework, What is express, Installation and setup of node js using express, Modules, Routing, Middleware, Controllers 	
Module 22) Structuring REST API	4
<ul style="list-style-type: none"> • Understanding the principles of RESTful APIs • Building CRUD operations with Express.js • Implementing middleware in Express.js • Implementing authentication and authorization in Express.js • Handling file uploads and other advanced features 	
Module 23) - Node - Integration with FrontEnd	3
<ul style="list-style-type: none"> • Integrating the server side node js API's with frontend • Testing the api's using Postman and Frontend 	
Module 24) Node - Professional Projects	8
<ul style="list-style-type: none"> • Build social network • E-commerce website • Blog Application • Canva website clone • Bookmyshow Clone • Olx Clone • AirBnB Clone 	